

Baze podataka

Vježbe 2

28. februar 2023.

Kreiranje tabela, primarni ključ i podrazumijevane vrijednosti. Select upit: **AND**, **OR**, **IN**, **BETWEEN**, **LIKE**, **IS NULL**, **IS NOT NULL**, aliasi kolona, izrazi u upitima, **UPPER ()** i **LOWER ()**, spajanje kolona. Izmjena i brisanje upisanih podataka.

1. Kreirati tabelu sa nazivom ZAPOSLENI i poljima:

- ID – numeričko polje, cio broj od 0 do 999. Polje treba da bude primarni ključ tabele.
- IME – ne može sadržati više od 20 karaktera.
- PREZIME – ne može sadržati više od 20 karaktera.
- STATUS – sa podrazumijevanom vrijednošću ‘stalno’, a druga moguća vrijednost je ‘honorarno’.
- DATUM_ZAPOSLENJA – tip date

Unijeti podatke ilustrovane Tabelom 1 u kreiranu tabelu. Pokušati unos podataka koji ne zadovoljavaju postavljena ograničenja polja.

Napraviti tabelu koja je identična tabeli ZAPOSLENI i koja sadrži svaki unos ove tabele. Tabelu nazvati ZAPOSLENI_N. Obrisati tabelu ZAPOSLENI_N iz zadatka pod a).

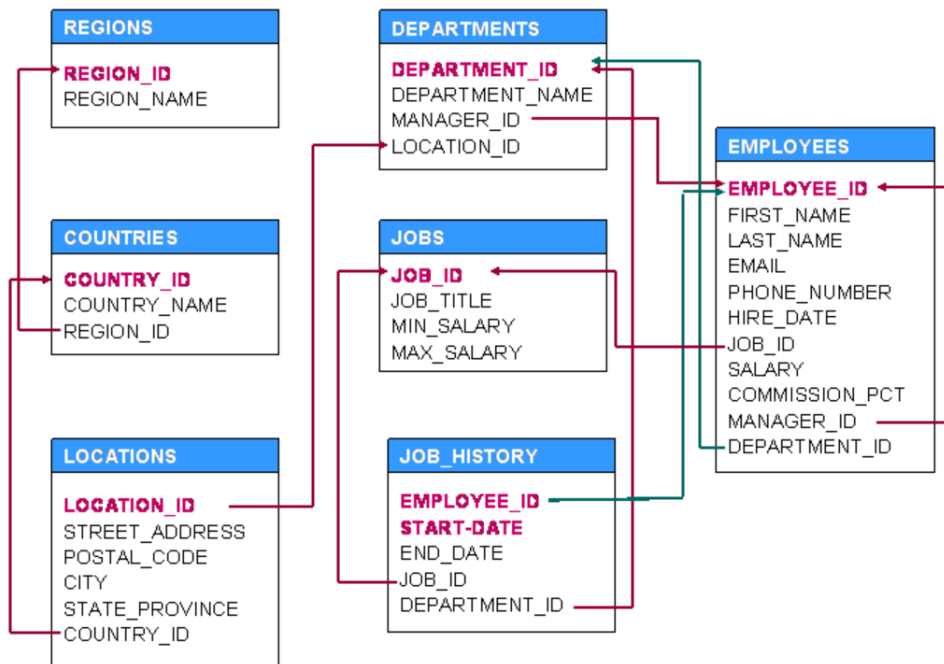
Tabela 1: Tabela ZAPOSLENI

ID	IME	PREZIME	STATUS	DATUM_ZAPOSLENJA
11	Marko	Marković	stalno	24. 03. 2001.
13	Petar	Petrović	honorarno	28. 09. 2012.

Posmatra se skup tabela (šema) korisnika HR, za koje imate pravo pregleda. Tabele koje se razmatraju su: EMPLOYEES, DEPARTMENTS, JOBS, COUNTRIES, REGIONS, LOCATIONS. Ispitati strukture ovih tabela. Obratiti pažnju na primarne ključeve, i strane ključeve (foreign key). U sljedećim zadacima potrebno je koristiti podatke iz ovih tabela.

2. Napisati upit kojim se iz tabele EMPLOYEES korisnika HR:

- prikazuju imena i prezimena zaposlenih i njihove plate. Nazvati odgovarajuće kolone prikladnim imenima (aliasi kolona);
- prikazuju imena, prezimena i plate onih zaposlenih kojima je plata u opsegu od 3000 do 7000 (zadatak uraditi na dva načina);
- prikazuju imena, prezimena i plate onih zaposlenih kojima je plata tačno 1000, 2000, 3000 ili 4000 (zadatak uraditi na dva načina);



Slika 1: Struktura tabela šeme HR

- (d) prikazuju imena i prezimena zaposlenih, i obračunata neto plata. Neto plata se dobija oduzimanjem poreza od bruto plate date u koloni SALARY. Porez se računa kao 10% od vrijednosti plate u ovoj koloni;
- (e) prikazuju imena i prezimena zaposlenih, i njihova plata, pri čemu ime i prezime treba da bude smješteno u jednoj koloni. Rezultat treba da sadrži samo one zaposlene čije prezime počinje slovom 'A' i slovom 'M', kao i zaposlene čije ime počinje riječju 'Alex'.
- Napisati upit kojim se izdvajaju zaposleni iz tabele EMPLOYEES (korisnika HR), čiji DEPARTMENT_ID ima vrijednost 50 ili 80. Imena i prezimena treba da budu zapisana u jednoj koloni, sa nazivom 'Ime i prezime', dok u drugoj koloni rezultujuće tabele, koju treba nazvati SK_ZARADA, treba da stoji treći korijen vrijednosti iz kolone SALARY zaokružen na dvije decimale.
 - Ponoviti prethodni upit, uz modifikaciju da budu prikazani samo oni zaposleni čije prezime počinje slovom 'A' i slovom 'W', kao i oni čije se ime sastoji od više od dvije riječi, pri čemu treba da budu sortirani po njihovoj zaradi u rastućem poretku. Zatim u rezultat uključiti i one zaposlene kojima je drugo slovo u imenu 'i' zajedno sa onima kojima je drugo slovo u imenu 'i' i predzadnje slovo u prezimenu 't'. Voditi računa da postoji mogućnost da se prezime zaposlenog može sastojati samo od dva slova, pa stoga treba i takvu mogućnost uključiti postavljeni upit.
 - Napraviti tabelu PROBA koja će predstavljati kopiju tabele EMPLOYEES korisnika HR. Korisniku čiji je EMPLOYEE_ID 110 promijeniti ime u 'T.john'. Napisati upit koji će u tabeli PROBA pronaći zaposlene kojima je drugi karakter u imenu '_'. Pomoć: koristiti klauzulu ESCAPE 'karakter' gdje 'karakter' predstavlja poseban simbol koji korisnik ORACLE-a bira u cilju pretrage koja će specijalne simbole '%' i '_' tretirati kao obične ukoliko se 'karakter' u LIKE dijelu nađe ispred nekog od njih.
 - Napisati upit kojim se:
 - Prikazuje spisak zaposlenih iz tabele EMPLOYEES korisnika HR kojima u koloni COMMISSION_PCT ima upisanih podataka.
 - Prikazuje broj telefona osobe za koju ne postoji MANAGER_ID (koja nema svog nadređenog).
 - Kreirati tabelu RADNICI u koju je potrebno upisati imena, prezimena i zarade za one radnike iz tabele EMPLOYEES korisnika HR, koji imaju zarade u opsegu od 2000 do 7000, pri čemu sva prezimena

treba da budu ispisana velikim slovima. Odgovarajući nazivi kolona treba da budu IME, PREZIME i ZARADA.

8. U tabeli RADNICI iz prethodnog zadatka promijeniti zarade koje su veće od 2000 i manje od 3000 na 5000.
9. Iz tabele RADNICI obrisati redove u kojima imena počinju slovom A, uključujući i one redove kod kojih je zarada u opsegu od 7000 do 7500.

Predlog rješenja

1. Kreiranje tabele:

```
CREATE TABLE zaposleni (  
    ID NUMBER(3) PRIMARY KEY,  
    ime VARCHAR2(20),  
    prezime VARCHAR2(20),  
    status VARCHAR2(9) DEFAULT('stalno')  
        CHECK(status IN ('stalno', 'honorarno')),  
    datum_zaposlenja DATE  
)
```

Pokušajmo nekoliko unosa:

```
INSERT INTO zaposleni (ID, ime, prezime, datum_zaposlenja)  
VALUES (11, 'Marko', 'Marković', '24-MAR-2001')
```

O formatu datuma biće kasnije više riječi.

```
INSERT INTO zaposleni  
VALUES (13, 'Petar', 'Petrović', 'honorarno', '28-SEP-2012')
```

Primjeri nevalidnih unosa:

```
INSERT INTO zaposleni  
VALUES (17, 'Proba', 'Proba', 'hororarno', '13-may-2021')  
INSERT INTO zaposleni  
VALUES (13, 'Proba', 'Proba', 'honorarno', '13-may-2021')
```

Zanimljivo je konstatovati da, uprkos **CHECK** ograničenju i definisanoj **DEFAULT** vrijednosti, baza dozvoljava unos **null**-a. Tehnički gledano, **null** nije vrijednost, već nedostatak vrijednosti, pa pretpostavljamo da je to razlog ove specifičnosti u sintaksi.

Kopiranje tabele i njenog sadržaja:

```
CREATE TABLE zaposleni_n AS SELECT * FROM zaposleni
```

Primijetimo da se ovim ne kopiraju ograničenja polja iz originalne tabele.

Brisanje tabele:

```
DROP TABLE zaposleni_n
```

2. Učitajmo tabelu EMPLOYEES korisnika HR

```
SELECT * FROM hr.employees
```

(a)

```
SELECT first_name, last_name, salary  
FROM hr.employees
```

```
SELECT first_name ime, last_name prezime, salary plata  
FROM hr.employees
```

```
SELECT first_name "Ime", last_name "Prezime", salary "Plata"  
FROM hr.employees
```

(b)

```
SELECT first_name "Ime", last_name "Prezime", salary "Plata"  
FROM hr.employees  
WHERE salary >= 3000 AND salary <= 7000
```

Drugi način:

```
SELECT first_name "Ime", last_name "Prezime", salary "Plata"
FROM hr.employees
WHERE salary BETWEEN 3000 AND 7000
```

(c) Prvi način:

```
SELECT first_name "Ime", last_name "Prezime", salary "Plata"
FROM hr.employees
WHERE salary = 1000 or salary = 2000 or salary = 3000 or salary = 4000
```

Drugi način:

```
SELECT first_name "Ime", last_name "Prezime", salary "Plata"
FROM hr.employees
WHERE salary in (1000, 2000, 3000, 4000)
```

```
SELECT first_name "Ime", last_name "Prezime",
       salary - 0.1*salary "Neto plata"
FROM hr.employees
```

(d)

```
SELECT first_name || ' ' || last_name "Ime i prezime", salary "Plata"
FROM hr.employees
WHERE last_name LIKE 'A%' OR last_name LIKE 'M%' OR first_name LIKE 'Alex%'
```

3.

```
SELECT first_name || ' ' || last_name "Ime i prezime",
       round(POWER(salary,1/3),2) sk_zarada
FROM hr.employees WHERE department_id IN (50, 80)
```

4.

```
SELECT first_name || ' ' || last_name "Ime i prezime",
       round(POWER(salary,1/3),2) sk_zarada
FROM hr.employees
WHERE department_id IN (50, 80) AND (last_name LIKE 'A%' OR last_name LIKE 'W%'
OR (first_name LIKE '_i%' AND last_name LIKE '%t_'))
ORDER BY salary ASC
```

5. Prvo kreirajmo kopiju tabele:

```
CREATE TABLE proba AS SELECT * FROM hr.employees
```

Prikažimo prvo red sa ID = 110:

```
SELECT * FROM proba WHERE employee_id = 110
```

Sprovedimo ažuriranje imena:

```
UPDATE proba SET first_name = 'T_john' WHERE employee_id = 110
```

Ponovo prikažimo rezultat:

```
SELECT *
FROM proba
WHERE employee_id = 110
```

```
SELECT employee_id, first_name, last_name
FROM proba
WHERE first_name LIKE '#_#%' ESCAPE '#'
```

Uočimo da karakter koji je stavljen iza ključne riječi **ESCAPE**, u ovom slučaju `'#'`, najavljuje da ono što slijedi iza njega u našem upitu (u našem slučaju `'_#'`) treba da se izbjegne, odnosno, ne protumači kao džoker (dio sintakse **LIKE** dijela)

6. (a)

```
SELECT employee_id ID, first_name ime, last_name prezime
FROM hr.employees
WHERE commission_pct IS NOT NULL
```
- (b)

```
SELECT employee_id ID, first_name ime, last_name prezime
FROM hr.employees
WHERE manager_id IS NULL
```
7.

```
CREATE TABLE radnici AS
SELECT first_name ime, upper(last_name) prezime, salary plata
FROM hr.employees
WHERE salary BETWEEN 2000 AND 7000
```
8.

```
UPDATE radnici
SET plata = 5000
WHERE plata BETWEEN 2000 AND 3000
```
9.

```
DELETE FROM radnici
WHERE ime LIKE 'A%' OR plata BETWEEN 7000 AND 7500
```